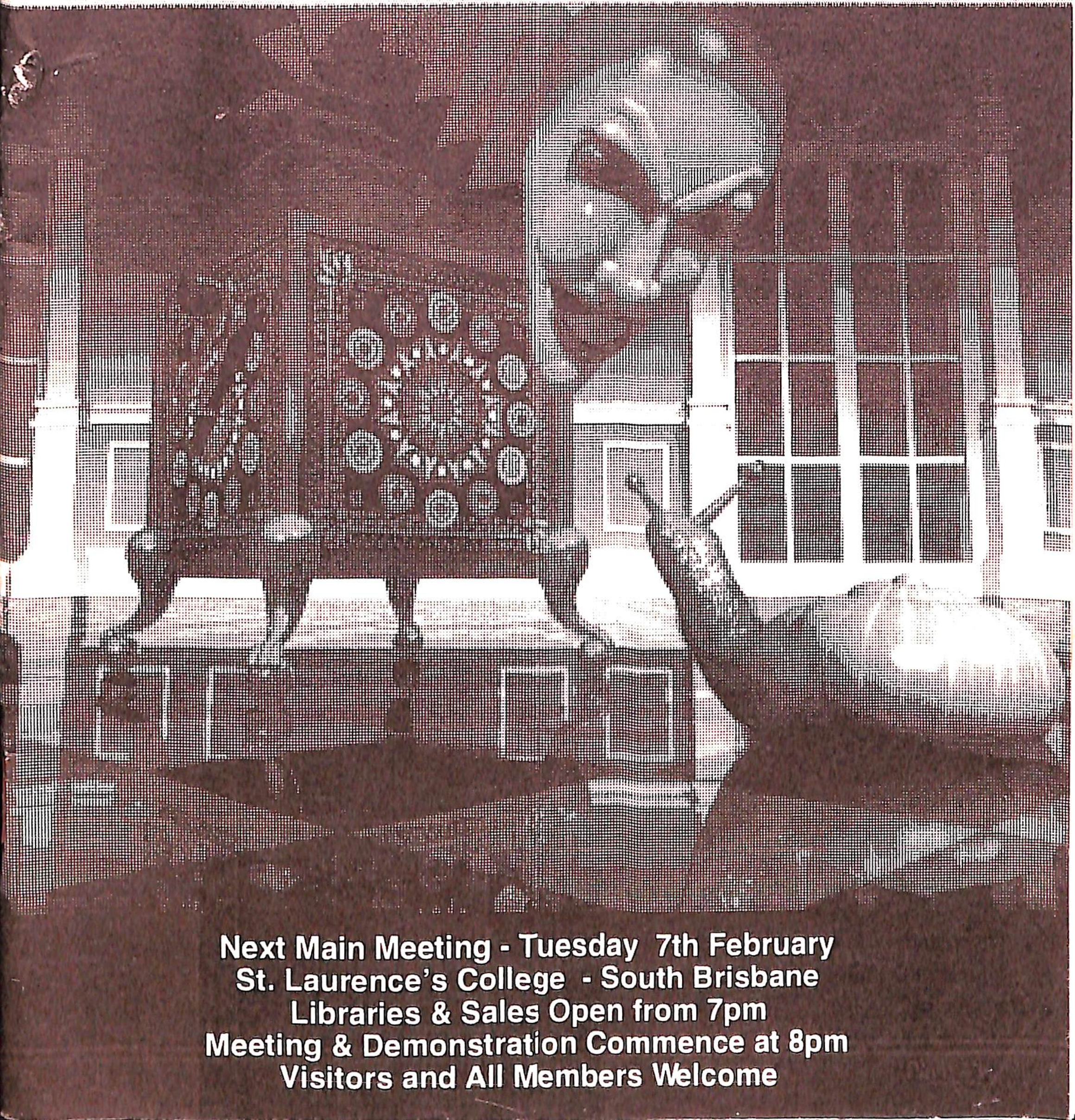
Queensland's Leading Amiga User Group Newsletter

CURSOR

COMMODORE-AMIGA
COMPUTER
USERS
GROUP
QLD Inc.

Print Post Publication No.
PP 432 356 / 00005

VOL.11 No.6 *FEBRUARY 1995*



Next Main Meeting - Tuesday 7th February
St. Laurence's College - South Brisbane
Libraries & Sales Open from 7pm
Meeting & Demonstration Commence at 8pm
Visitors and All Members Welcome

WHEN THE CHIPS ARE DOWN
IT'S TIME TO SEE

The
AMIGA
Specialist
Tony May

ALL WORK
GUARANTEED

Voice (07) 279 5503
Fax (07) 279 5507

For Your Convenience
Goods May Be Left At

Megabyte Computers
Home Base
34 Goggs Road
Jindalee

SERVICE &
ATARI
AMIGA
& PC
UPGRADES

South East Qld Computer Repairs

MANAGEMENT COMMITTEE

PRESIDENT:	**Keith Antoine** Ph. 300-2161
SECRETARY:	**Barry Benyon** Fax or Ph. 290-1521
TREASURER:	**Bob Baxter** Ph. 264-3702
LIBRARIAN:	**Steve Knight** Ph. (074) 468-469
NEWSLETTER EDITOR:	**Mike Williams** Ph. 801-1020

C-64/128 CONTACT:	**Doreen Horne** Ph. 398-7441

B. B. S.s

We have two BBSs available for C.C.U.G.Q. members:-

SideCar Express BBS ... 3 lines
(07) 805 5928 ... 3 lines on the one number
(075) 463 252 ... local call from GoldCoast
SYSOP: **Brendan Pratt**
Voice: Ph. (075) 463-317
FidoNet Node No. 3: 640/463.0
AmigaNet Node No 41:400/463.0

WindDragon Inn BBS ... 3 lines
(07)300 6357; (07)300 6356; (07)300 5569
SYSOP: **Keith Antoine**
Voice: Ph. 300-2161
FidoNet Node No. 3: 640/562
AmigaNet Node No 41:400/556
These two BBSs are available at speeds of 300-28,800 baud

Mailing Address:

The Secretary
C.C.U.G.Q. Inc.
P.O. Box 274
SPRINGWOOD QLD 4127

Commercial Advertising Rate:

$40.00 per full page, per issue
$25.00 per half page, per issue
This rate is for camera-ready A-5 pages.

Opinions expressed in CURSOR articles are those of the Author(s), and not necessarily those of the C.C.U.G.Q. or the Editor. Permission for reprinting by other Users Groups is granted, provided that both the source and author are acknowledged.

CONTENTS

South East Qld Cmp. Repair Ad. 2
Committee Members.................... 3
B.B.S.s.. 3
Next Meetings.............................. 3
President's Patch.......................... 4-5
Coming Meetings.......................... 6
The Cover Story............................ 6
Megabyte Computers Advt........... 7
The S:Startup-sequence 8-9
Editorial...................................... 10-11
Beginners 11
Questions & Answers 12
Aliases I Have Used 13
What's In The Mags 14-15
B.B.S. News............................... 16
Imaging Technology Advt...........17
Mike Sinz's Amiga History 18-19
Hardware Story - SCRAM500 ... 20-21
Design Philosophy 22-23
Quick Reference Card for ED..... 24-25
Amiga Public Domain Disks....... 26-30
Parcom Advertisement.................41
Club Members Advertising......... 43
Megadisc Subscriptions 47
Public Domain Disk Details........ 47
New Club Membership Fees....... 47
Membership Application Form... 48

MAIN MEETING

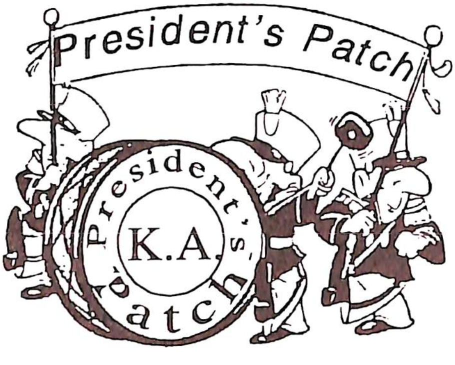
The next meeting will be held on
Tuesday **February 7th** (7pm - 10pm)
St. Laurence's College
82 Stephens Road
South Brisbane Qld.

WORKSHOP MEETING

The next workshop meeting will be held on
Sunday **February 12th** (11am - 4pm)
Ithaca R.S.L. Hall
Cnr Nash & Elizabeth Streets
Rosalie Qld.

Published by:
C.C.U.G.Q. Inc.
P.O. Box 274
Springwood
Qld 4127

Printed by:
Towns-End Printing
P.O. Box 1333
Beenleigh Qld. 4207.
Ph. (07) 807-0177



Oh dear! Not again! It seems like only yesterday I was doing this, and 2 months have gone past.

C=!!!
Its really no use swinging abuse, or telling others what should be done, or even reading all the rumours that are flying around, for we are all in the hands of Liquidators and Lawyers. They could not care less that you and I want Amiga to arise phoenix-like from the ashes. All they care about is big fees and getting what they can for the bare bones of C=, for the creditors.

The news about C= is very decisive. "There isn't any" except the odd rumour about CEI and C=UK being told their offers were rejected by the liquidator once again, and then CEI saying the rumour is not true!!

I have some news on C= though, and it is that Ross Delaforce has in his posession a video tape, made by Dave Haynie of "The Last Day of Commodore US" and the subsequent party. He is bringing along a video projector and screen, plus I will bring audio speakers, so it should be a good night at the February meeting at St. Lawrences.

By the way, it will not be held in the hall but in a classroom at the upper level.

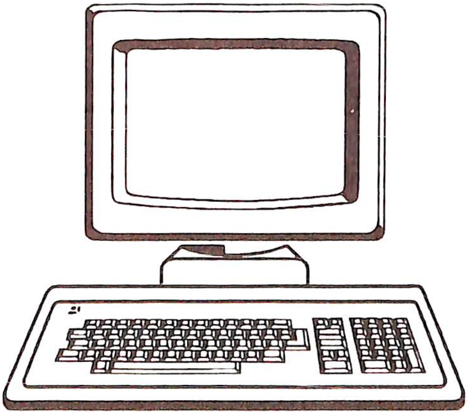
There is some new software available in the Commercial Library (run by Steve Knight), some of which might interest you:-
Imagine 2.0, 2.9 and 3.0
Blitz Basic
Dpaint III and IV (more copies)
Fresh Fish CD-Roms
Amiga Magazines (latest issues).

The committee is trying very hard with limited funds to keep things up to date and interesting.

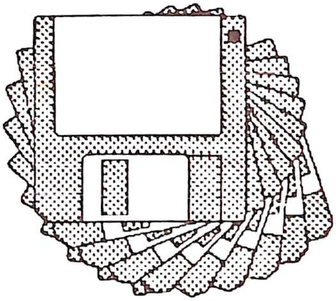
One thing that I need from members is some assistance with the meetings at St. Lawrances on the first Tuesday of each month. As there are so few developers or anyone of professional standing in the Amiga world available to us to give talks these days, I am therefore asking members to give of some of their time. This would entail a half hour talk on their pet subject or one of expertise. If no members volunteer I can forsee us shortening the Tuesday nights in the near future. Give me some feedback on your thoughts on this subject.

I trust that everyone had a good Xmas and New Year and got all the good presents that they wanted. I have noticed that there are some faster modems around on the BBS. All of a sudden a member speeds up--ZZZOOOOOOMM.

I will be talking about the BBS and my graphics station in the BBS Article but I did not have the Xmas I wished. What did happen was my two monitors went "down".



One fried and the other just did not work. And a 3000 main board gave up the ghost, its now spare parts. (Anyone who needs spares for his 3000 give me a ring!)



I can also give you a new repairer (for Commodore Amiga Monitors especially). These repair houses are few and far between as repairing a monitor is totally different to repairing a computer or a TV. Not only that but they also have the parts.
ATS Computers (Australian Tech Support) 700 Gympie Rd, Lawnton. PH. 881 3010 or 2087 and ask to speak to Tom or Andy. They can also do some repairs to Amiga computers, but if they cannot they will say so, and there is no charge. The average cost to repair a monitor is $100, so think carefully before repairing a 1084, but its worth it for a multiscan.

Till next time,

Keith Antoine

"El Pres"

Coming Meetings

Please always refer to the latest CURSOR for information on the next meeting as the following plans are always subject to change.

February 7th 1995

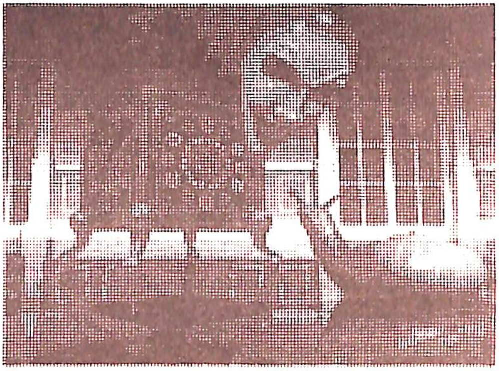
Video by Dave Haynie

Ross Delaforce will present a video, made by Dave Haynie, Amiga Guru, of the "Last Day of Commodore US" and the subsequent party.

PLEASE NOTE:

This and all future meetings will be held in the classrooms at the upper level of St. Laurence's College and NOT in the Auditorium.

The Cover Story



The cover picture this month is a 256 colour image called "Jack6". It was sourced from a CD-Rom titled World of GIF which contains over 3000 high resolution pictures, most of them in the GIF format.

The file is a 800 x 600 image of 227,069 bytes.

The picture was scaled by 54% horizontally and 61% vertically to fit the page (using Pagestream).

A test print revealed that the image did not need any "massaging" through Art Dept Professional and is a direct printout of the image through "PageStream".

Lots of different interpretations could be made of this picture, e.g. the Jack-in-the-box could represent the Amiga surprising IBM (the snail) who thought the Amiga was dead and buried.

Megabyte Computers

Your Complete Computer Specialist

Enough Said

A QUEENSLAND ENTERPRISE

Amiga, IBM Compatibles, Mac, Hewlett Packard, Canon, Citizen, GVP, Electronic Design, Roctec, Software Solutions, CD 32, Multi Media, Scanners, Fax/Modems, Repairs & Upgrades, Real Support, Real Service, A Real Computer Store

and proud of it!

PH:07 2790666 OR 008 8180716

MEGABYTE COMPUTERS
HOMEBASE JINDALEE
JUST OFF CENTENARY HWY

The S:Startup-sequence and how to change it

by Chris Eburn

The S:Startup-sequence is a list of DOS commands executed by the Amiga on startup or Warm-Boot. It is in every way a normal BATCH file except for its name. A batch file is a collection of DOS commands that are executed as if they were typed at the CLI/Shell prompt.

It must be said that Batch files can do more than typed commands as they accept variable names and have the ability to apply logic tests with user interaction. Batch files are a simple(?) programming language of which the startup-sequence is one of the more simple examples.

To view and/or change the startup-sequence you must use an ASCII viewer (nearly any example will do - e.g. "More") and an ASCII editor. A batch file must be ASCII so some wordprocessors are unsuitable, if in doubt use ED.

ED is a screen based editor supplied by Commodore with the system software giving it the advantage that everyone has it and the documentation to use it (it also works and is small). *See pages 24 & 25 for a list of keyboard commands.*

To load a viewer so that you can read the startup-sequence without danger of changing it I think is a good idea. The method of loading is similar to loading an editor and gives you practice with the Shell.

A simple method is:-
Find and double click the SHELL icon (on the sys BOOT disk).
Enlarge the window to fullsize and type at the prompt:-

```
CD s: <return>
More startup-sequence <return>
```

The <return> means press the keyboard RETURN key.

The startup-sequence will now scroll up your screen. A better viewer gives better control but the method of access will vary and is up to you to work it (press H for the MORE Help menu).

If all this is clear then working with a COPY of your workbench:
open a SHELL (double click icon) and in this shell enter

```
cd s: <return>
ED startup-sequence <return>
```

NB: to Quit ED without saving changes:
press esc
then Q
then <return>

The 1.3 S:startup-sequence will look something like this:-

```
c:SetPatch >NIL: r ;patch system
```

functions

```
Addbuffers df0: 10
cd c:
echo "Amiga Workbench Disk. Release 1.3.1 version 34.25"
Sys:System/FastMemFirst ; move C00000 memory to last in list
BindDrivers
SetClock load ;load system time from real time clock (A1000 owners should replace the SetClock load with Date
FF >NIL: -0 ;speed up Text
resident CLI L:Shell-Seg SYSTEM pure add ;activate Shell
resident c:Execute pure;This lets resident be used for rest of script
mount newcon:
failat 11
run execute s:StartupII
wait >NIL: 5 mins ;wait for StartupII to complete (will signal when done)
SYS:System/SetMap usa1 ;Activate the ()/* on keypad
path ram: c: sys:utilities sys:system s: sys:prefs add
;set path for Workbench
LoadWB delay ;wait for inhibit to end before continuing
endcli >NIL:
```

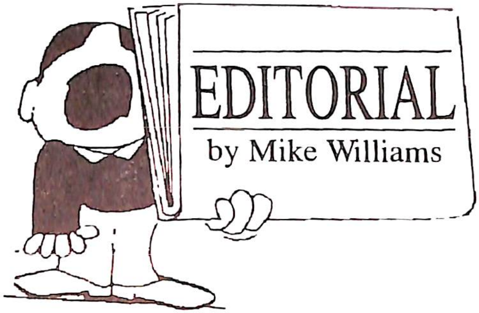
This sequence calls and executes a second file called s:StartupII.

```
resident c:Resident pure
resident c:List pure
;pre-load LIST and CD
resident c:CD pure
resident c:Mount pure
;the next 3 are loaded for speed during startup
resident c:Assign pure
resident c:Makedir pure
;make IF, ENDIF, ELSE, SKIP,
;ENDSKIP, and ECHO resident if
;you use scripts much, and can
;afford the ram. also make Failat,
WAIT, and ENDCLI resident if you
use IconX a lot.
makedir ram:t
assign T: ram:t
;set up T: directory for scripts
makedir ram:env
; set up ENV: directory
assign ENV: ram:env
makedir ram:clipboards
;set up CLIPS: assign
assign CLIPS: ram:clipboards
mount speak:
;just mounting doesn't take much ram at all
mount aux:
mount pipe:
resident Mount remove
;if you have enough ram, keep these
;resident
resident Assign remove
;by removing these lines resident
Makedir remove
break 1 C
;signal to other process its ok to
;finish
```

Both of these files together are the standard Amiga startup. These files are long and complex so that they can cope will all types of Amiga environments from the original 1000 to a machine with expanded memory, harddrives, multiple floppies, clocks etc. etc.

It is this generality which allows us room to personalize our own startup for our own needs and wants.

Continued on page 30



EDITORIAL

by Mike Williams

Hello again, Club Members and welcome to this, the first issue of Cursor for 1995.

I have had a nice little break (from creating the Cursor) for a few weeks over the Christmas/New Year period but I'm now looking for articles from members for the next issue. Remember, the quality of each Cursor is purely dependant on your input to it. If there is no input ... there is no Cursor.

Don't leave it to someone else to do, write a short article yourself on what you like about the Amiga, or how you use it yourself.

In this issue of the Cursor, we have some more articles from our Tasmanian Correspondent, Chris Eburn, and a very informative and in-depth reply by Neville Duguid to our first question in the "Question & Answer" column. I also received a second, briefer answer to Derek Hackett's problem, sent in by Paul Edwards, which is also reproduced. *Thank You, Paul and Neville.*

We also have an article from Mike Sinz, who was in charge of the Amiga Operating System with Commodore during the final years. The article is mainly about the Amiga 3000 and gives us an insight into what went on in Commodore US during this time.

We also have another rare article from Norman Jackson of MegaMicro Technology in Sydney. Norman designed the SCRAM500 expansion kit which is described by our Tasmanian Correspondant elsewhere in this magazine. In his own article, Norman describes his design philosophy when creating the SCRAM500 and goes into some depth as to how it was conceived. It also provides some insight into how the Amiga itself is designed.

We also have details of new Beginners Classes by our President/BBS Sysop, Keith Antoine.

Please also note that there is a change of venue for the Tuesday Night Main Meetings. We are still holding them at St. Laurence's College, but instead of in the Auditorium, they will now be held in the classrooms at the top level of the College grounds. Members may recall that the November meeting was held there because the Auditorium was double-booked for school plays.

The Committee has decided to have all meetings from now on in the classrooms for a few reasons:-

1. It is much cheaper to hire the classrooms (next to nothing!).
2. It is a friendlier atmosphere in the smaller rooms, rather than the big impersonal Auditorium.
3. It is less imposing for speakers to "show their stuff" in a classroom rather than on a stage.
4. The idea was well received by club members when suggested at the previous meeting.

Talking of our meetings, if you have already read the "President's Patch" you would see an appeal by Keith for club members to come forward and give a short demonstration on how they use their computer. This demonstration would only be for about half an hour and we would probably have two different members presentations at each Tuesday Night meeting.

As explained by Keith, these presentations are necessary to "tide us over" during this uncertain period, until the Amiga resurfaces again and we can find professional people to continue on the demonstrations.

I urge all members to give it serious consideration.

Happy Computing.

Mike

Mike Williams - Editor

Beginners

With the permission of the committee I have had talks with Bob O'Neill of Care services Inc. and they have agreed to let us use, free of charge, their premises in Ferny Grove for beginners classes.

The first of such gatherings will be on the 19th February from 10am till 2pm. This will be fairly informal till we can find out exactly that which most attendees require to learn.

There will be a door fee of $2.00 each to cover sandwiches tea/coffee and bikkies. We will need feedback as to what it is that is required i.e communications packages-modeming/bbs/internet etc. Graphics Dpaint,Lightwave,3D rendering, Wordprocessing etc.

The address is McGinn Rd, Ferny Grove and is two-thirds up the road from Samford Rd, and on the R/H side in a shopping centre which actually backs onto McGinn Rd.
By the way if anyone feels that they know sufficient about a subject and will be willing to help out contact me on 300-2161.

See you there.

Keith Antoine

Questions & Answers

In Novembers Cursor, a question on startup-sequences was posed by Derek Hackett. Here are two answers: A brief one by Paul Edwards and an extensive one by Neville Duguid. Both of these make very entertaining reading for all members.

Mike, my name is Paul Edwards. I have been a member for about 15 months now and you probably remember me from the Sunday Workshops. I have an Amiga 600 with a 65meg Hard Drive and 2 megs of ram. At the moment I am trying to get a modem from the library, so I can try before I buy one. I think this is the way of the times: info and more info. Out my way (Redbank) the council is starting a network up for the schools around here. They gave them a modem for free plus free use of the network until Chistmas. Kind of them to throw my rates $$$$'s around. The network is to our sister city in Japan and they hope to swap info. and cultural ideas. I will be able to check it out soon, I hope?? I am glad the Sunday Workshop has been extended to start at 11am - I always feel rushed on Sundays.

Concerning a name change for the CCUGQ, what do you think of QUAG???? Queensland United Amiga Group QUAG.???????

This is my first attempt at writing an entry for Cursor. I am trying to help a Derek Hackett with his Amiga 600 HD shell start-up file.

`prompt "%n.%s>"` = default shell ie: 1.sys:>

Aliases are only for the shell you are in. Aliases=short for the complete command or replace it with a different name.

The next two lines are a loss to me??

```
alias normal "echo *
```
???? Cannot get argument to work.

```
Alias reverse "echo*
```
???? same again

```
alias clear "echo
```
means echo to shell window.

```
alias emacs memacs
```
means print emacs at the shell prompt and shell loads memacs

```
alias xcopy"copy clone"
```
means the copy command now for the start-up sequence:

Concerning the 2 different startup-sequences, one s:start-up sequence is all that is needed. It looks as thought someone has added this later (hd.file) because you do not need part of this sequence.

Unless you have a file/dir/or a disk called memos (???) you do not need this "s:start-up sequence" at all.

Paul Edwards.

"Aliases I Have Used"

by Neville Duguid

In response to Cursor 'Questions & Answers', November, 1994.

Two questions from Derek Hackett. This month Question 1. Next month Question 2.

Question 1. " The following is my shell startup file. What does it all mean?

```
alias emacs memacs
alias xcopy "copy clone"
alias clear "echo *"*E[0;0H*E[J*
                                " "
alias reverse "echo *"*E[>1m*E[3
               0;41m*E[0;0H*E[J*""
alias normal "echo *"*E[>0m*E[31
               ;40m*E[0;0H*E[J*""
prompt "%N.%S> "
```

Derek, I am glad you asked.

The Amiga Shell is my preferred working environment, and "Alias" just happens to be my favourite command. I have been thinking of writing an article about it for some time. Your question has given me the excuse.

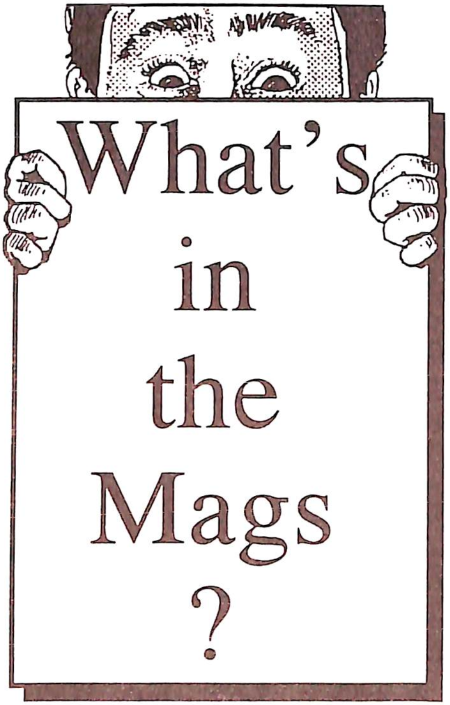
"Alias" and "Prompt" are AmigaDOS commands, and "Shell-Startup" is an AmigaDOS script. AmigaDOS commands are programs normally "resident", or kept in the Amiga's C: directory. They are executed by typing their names at the Shell prompt. As an alternative, a Script File may contain a sequence of commands which are executed as a batch. You can create or alter script files with a text editor like "Ed", which you will find in your C: directory along with most of the other commands.

Otherwise, to find out what the "alias" command does interactively, double-click on the Shell icon (usually kept in the System drawer). When the Shell window opens, type "alias" (without the quotes) and hit the Return key.

When "alias" is not followed by anything else on a line, it responds by listing all aliases currently in effect for the Shell you are using. (These would typically correspond to the examples in your Shell-Startup file). Aliases are a type of variable - names that are assigned other meanings. They are a shorthand way of referring to more complex AmigaDOS commands, or may be used simply to personalise your working environment.

The first example in your Amiga's Shell-Startup clearly belongs in that category. The Amiga comes with Micro Emacs ("MEmacs") which you will find in your Workbench Tools directory. It is a smaller version of the Emacs that goes with Unix. (Unix: a much larger operating system also available for sufficiently expanded Amigas). I suspect the software engineers at Commodore may have been in the

Continued on page 32



This is the second edition of our new column which aims to give the reader a quick overview of some of the more interesting articles or reviews seen in the latest Amiga magazines.

As this edition of CURSOR covers a period of two months, you have a bumper crop of nine magazines reviewed here this month.

All of these magazines (and more) are available for borrowing from Steve Knight, our Commercial Librarian at our Main Meeting held on the first Tuesday of the month.

Here is a sample of the articles appearing in these magazines.

AUSTRALIAN COMMODORE & AMIGA REVIEW (A.C.A.R.)

- January 1995

1. A review and comparison of two removable storage hard-drives, the Bernoulli and the Fujitsu.
2. CEI boss Alex Amor speaks out on his strategy for Amiga (if his bid is sucessful).
3. A 5-page review of different software for "LightWave" which is now available in Australia.
4. Using your Amiga on the Internet. This article shows what it's all about.

AMIGA SHOPPER

- October '94

1. Review of WorkBench 3.1 ... The title "Upgrade or die!" says it all.
2. A supertest comparison of 10 different A1200 accelerators.
3. A review of an A4000 accelerator called "Warp Engine".
4. Reviews of the latest CD-Roms including FreshFish.
5. Amiga Answers - 8 pages of advice and solutions to readers questions and problems
6. Tutorials on Video, AmigaDOS, Amos, Music, Assembler & Chess.

AMIGA WORLD

- November 1994

1. Video Toaster Special including 15 tips for mastering the Toaster to create more effective video productions.
2. The Joy of Fax - everything you need to know about turning your Amiga into a full-service Fax

Machine.

3. Macintosh emulators compared. A-Max IV vs. Emplant.
4. Review of the Power XL High Density disk drive.
5. Three High-Resolution multi-scan monitors compared.
6. Game Review of the Adventure Game "Kings Quest VI".

AMIGA WORLD
- December 1994

1. Imagine 3.0 tutorial (3D graphics software).
2. FinalWriter 2 - 4 pages of hint and tips in the form of a tutorial.
3. Review of TypeSmith 2.5 the font-management software from Soft-Logic.
4. A review of PC-Task, the low-cost IBM emulator.

AMAZING AMIGA
- October 1994

1. A review of Bars & Pipes Professional 2.5 (Music program).
2. An exploration of the latest 3D rendering software, Imagine 3.0
3. C.E.I. President advises his plans, if he is sucessful in purchasing the Amiga.
4. A 3-page tutorial on using Aladdin-4D.

AMAZING AMIGA
- November 1994

1. A review of Distant Suns V5.0
2. A Personal Information Management program "On The Ball" is reviewed.
3. Adorage and ClariSSAare reviewed. Only one of them gets the "thumbs-up".

AMIGA USER INT'NATIONAL
- October 1994

1. Tests of two external hard drives for the Amiga 1200.
2. The Zappo CD drive for the A1200 is reviewed.
3. A review of the Vortex SLC2 Bridgeboard which is a full 486 on a card running at 50Mhz.

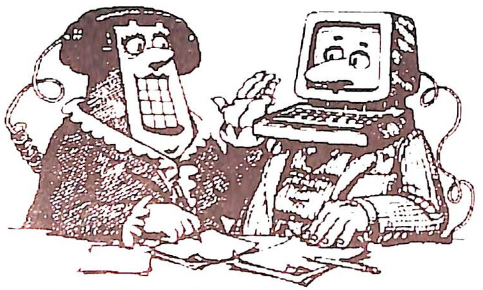
AMIGA USER INT'NATIONAL
-November 1994

1. A review of "Part and Particle" (commonly called PAP) which is an animation program for use with OpalPaint.
2. An inkjet printer "Texas Micromarc" is put through its paces.
3. A review of a dedicated Amiga CD-Rom called "EuroScene" which has over a gigabyte (1000 megabytes) of Demos, Sound modules and Graphics.
4. A Mega-Review of 7 different Ram-Expansion and Accelerator boards for the 1200 (with a couple of Hard-Drives thrown in for good measure.

AMIGA COMPUTING
- December 1994

1. The spreadsheet, TurboCalc V.2 is reviewed.
2. ImageMaster, a rival for ADPro or ImageFX shows its stuff.
3. ProText 6.5 is the purists word-processor is reviewed showing

definite advantages over the opposition in speed and memory usage.



B.B.S. News

WindDragon Inn

Well, the BBS is still on a 3000, but what a 3000. Due to failure of a motherboard and cash flow problems, I decided to put both into one box, Graphics and BBS. So what I now have is: 3000/ 040/33 Warp Engine 34mb Ram Micropolis AV2010 1.08gig video streaming HD, Micropolis 4010 1.08 gig HD for Aminet Files and internet News, Seagate 1.8gig HD for BBS files and a Quantum 540 mb HD for all running programs, plus 3 CDRoms online. There is also a tape backup unit too and an external 1.44 floppy. Then the cards:- V-Lab motion, EGS, Serial card for the modems and Toccata 16bit sound card.

OK, I can hear you say that's too many for a 3000 scsi. Well that's true except that the Warp has a scsi II device which is running in tandem with the normal scsi.device, so we have 14 scsi addresses. I will also be setting up a 486 DX2 66 with OS/2 and LoraBBS in February. This will give us another 3 lines 1 more for internet and 2 for access, don't worry you can access the Amiga files via this one too, and I hope the access to internet will be better via the IBM. We will also later on add an upmarket BBS software for conferencing Fax online Rip style graphics and databases.

I am also going to buy a Canon BJ 600 colour printer for A4 style colour hardcopy output, anyone interested in its output see Rudy Florentinus at one of the Sunday meetings, he has some amazing examples he has printed out.

Keep an eye out for the small article dealing with begginers classes, bearing in mind that we can all learn something from someone, and that these gatherings are more of a socialise and learn setting rather than the strict learn something arena. So please come along and support the effort and do not be afraid to spread the good word outside the realms of the club.

So behave youselves (if thats possible) till the next meeting.

Keith

IMAGING
TECHNOLOGY

RE-MANUFACTURING LEADERS

Printer Ribbons
Inkjet Cartridges

SUIT ALL MODELS

Also

Complete Range of NEW Products

BUY DIRECT FROM THE MANUFACTURER

LASER TONER CARTRIDGES

Complete Dismantle And Reassembly

Replacement Drum
Guaranteed 10 Refills

Savings of approx 40 %

PRIVATE AND COMMERCIAL SUPPLIERS

AUSTRALIA WIDE 008 077347

FAX (07) 856 4949
PHONE (07) 856 4355

80 WEBSTER ROAD STAFFORD QLD 4053

MIKE SINZ'S VIEW OF THE AMIGA'S HISTORY

Mike Sinz was in charge of the Amiga OS group during its final years. Here is his view of some of what went on during that time, reproduced here with his permission.

I think the reason many of us see the A4000 as a hack is because in engineering terms it really was a hack. Other than the AGA chips. (AA for those who actually worked on them).

The A3000 was a new motherboard design with new bus designs and chips to support and integrate all of this. The A3000 had a number of radical departures from earlier designs *other than* the ECS chipset. I won't go into all of these here but they were significant and many were leading edge. (Again, the chipset did not change so that part was not "new").

The way the A4000 became really shows this. Now that the story is public (in more than one place) it can be talked about. Remember, there was a changing of management just as AA technology was being completed. The original management was pushing two systems: The A3000+ and the A1000+ The A3000+ was an AA (oops, AGA) based machine with the same box/formfactor of the A3000 but also included an AT&T DSP32 (very cool) and was very tightly integrated. (SCSI, DSP, AGA, etc.) The DSP system was very cool and the DSP was powerful enough to do a full V.32bis/V.42bis FAX/Data modem *IN SOFTWARE*. (The 2400 baud software was even scheduled to be included with the system - the V.32 stuff cost too much to include for "free"). It would also do really good speach and sound and math (talk about fast rendering times!).

The A1000+ was a super-low-cost 2-slot AGA-based Amiga. Goal was to have this thing list at $999 including hard drive, 2 or 4 megs of RAM depending on RAM prices, keyboard, etc. It would have been about the size of the A1000 but without the keyboard storage area (no room for it). Really nice system. (And it was going to be either 14MHz or 28Mhz 68020 or 68EC030 system).

Anyway, what happened was first that both of those systems were cancelled and the A600 was built (an ECS system that cost more and did less than the A500).

Then, they tried building the A1000jr (our internal nickname) which was an ECS A1000+ design but in a different casework that ended up making the machine cost much more to produce and do much less than the A1000+.

The marketing group killed this. (Yes, they were just starting to get to know what they really wanted and were getting sales to grow...). Well, in response to this, they took the A1000jr design (which was a really hacked up A3000 design in a bad case) and made the A3200 design which was AGA based and very hacked A3000 motherboard chipset in a bad case.

The case was so bad that they had to put the CPU on a card in order to make it fit. (This later was turned around by marketing to be a benefit, except since the bus design of the CPU card was never really reworked for this design, only 68030 CPUs were easy to make work and 68040s were rather poor due to the 68030 bus protocol required.). Anyway, the cost was so high that they pushed the machine up as the A4000 and shipped it with the very nice 68040 processor (on a rather limiting CPU card) to justify the cost. They had removed DMA disk I/O and replaced it with PIO IDE. Thus, you could not even connect a CD-ROM to the new Amiga which was billed as the best multimedia machine. They even made the casework so you could physically put a CD-ROM into the case but the motherboard could not support it so... Anyway, the A4000 finally shipped as a product and it got there from a series of major management choices that just caused the product to be worse than it needed to be or cost more than it needed to.

The engineers involved did their best to deal with the choices management made. Some things on the A4000 run a bit better due to this. (Such as the better buster for Zorro-III support and switching to standard 72-pin SIMM memory modules, etc.) But these were really just "slipped" in rather than part of the grand design. (It was a very frustrating time at Commodore).

Many of us still cringe over the silly choices the Management made during that time. They really cost Commodore a ton. If it would have saved Commodore to not have made those choices is not clear but Commodore would not have died as quickly, that is for sure.

Anyway, sorry for this long thing. I have left out many details and arguments we (the engineers) had

Continued on page 40

Hardware Story
SCRAM500
by Chris Eburn

Another article from our Tasmanian correspondant (from the Burnie Amiga Users Group).

My Amiga is an A500 with 3megs of memory and a SCSI Harddrive. This expansion was achieved through the medium of a SCRAM500, (SCsi and RAM for the a500) by MegaMicro Technology. This unit was bought in kit form as an expansion for an A1000 providing an expansion path for either machine (1000/500) to 8megs of RAM, AutoBoot harddrive, Tape drives, CDrom and other SCSI bus devices. This is the story of purchase and kit construction.

The choice for the SCRAM was based on cost, local supply and the excellent write ups I had read elsewhere. My first contact with the project itself was a public domain disk that held all the assembly /parts details with IFF pics of the circuit boards and final product. These documents point out that the board is of very high density making it very hard to produce at home and that three of the controller chips are MegaMicro copyright which you must buy from them. The kit thus provides for several levels of entry, you can buy only the custom chips and make or supply all the other components. You can buy the finished unit in its case ready to go, OR any combination in between. The choice should be based on your solder skill and your ability to obtain the other common components.The SCRAM is NOT a beginners first project but is not as hard as the documents suggested.

Once I was convinced that the 1/2meg A1000 needed to be expanded, that the SCRAM500 could be the answer and I had the skill to attempt such a project I rang MegaMicro (02 980 6986). The phone was answered by Mark Ireland who gladly answered all questions about the cost of various kit and memory options, SCSI with no ram or 256k to 2/4/8Megs. Mark proved more than willing to talk on a range of topics and has provided technical information on other occasions. Mark made the point that the SCRAM was designed for the A500 and may not work on the 1000, he had used them on some OK but others required a higher quality processor (it would also be pointing backwards as the bus while identical is on the other side of the machine). I took the risk and decided on 2megs Full kit and quickly sent payment. I will not quote cost as these always change with time and I have had the SCRAM for a year now.

A couple of days later a parcel had arrived for collection at the local post office. I rushed out to return with a medium sized padded bag. I opened the bag and hoped I hadn't made a huge mistake. On second inspection I discovered a very neat circuit board, well packaged range of components, a well finished case (designed to suit the A500) and a hard copy of the instructions. I read all of this book twice and then again making notes. The key it seemed would be time, good light and solder skill plus the care associated with all intergrated circuits.

I began ..... the memory sockets were the hardest, row upon row of high density pins. The SCRAM uses ZIP dRAM to produce a very compact board. While the ram area needs care it is not as hard as the manual and first inspection suggested. The remainder of the board is easy for anyone with kit experience. I used sockets for all chips at extra cost but at greater safety and peace of mind (can't burn chips that aren't there). A couple hours work saw the finished IC empty board that had been checked visually and electrically for shorted or broken tracks, bad solder joints and component orientation. And then I checked it all again.

Next step is to install the common (normally soldered) chips into the sockets and checked it all again .... OK!? Next test is called the smoke test. Plug the unit into the side of the computer and power on (if smoke comes out you have done something wrong). The trusty 1000 showed its little hand, test ONE OK, I took another breath. At this point I will say that the SCRAM has three lights on the front --- RED ram in use --- YELLOW scsi in use --- GREEN system has booted OK.

Test 2 is to install the SCSI side of the board plug in switch on, watch for more smoke. The little green light came on AND the Amiga put up its hand, (NO yellow as I didn't have a harddrive yet) take another breath.

Test 3 is the same again with RAM chips though by this stage I was more confident. The 1000 crashed in a big heap!!! I rang Mark, and after discussion decided that my A1000 had the cheep 68000 processor I would need a replacement (it had recovered after the SCRAM was removed, no smoke).

This more rare processor took a couple days to organize and is a story of its own, I had to buy two. After the brain op, back to step 3. SCRAM in, Power on and yes Green lights! Red lights! System is go. I loaded workbench and asked the CLI -AVAIL- to be told 2.5Megs we had expanded the AMIGA 1000 to undreamed

Continued on page 42

DESIGN PHILOSOPHY
SCRAM500
by Norman Jackson

Some years ago I designed a product for the Amiga which was a very simple 2 Meg RAM board for the A500/A1000. I was amazed at the success of this little project and decided to do a sequel with more features - but still simple and cheap.

8 Megs of RAM and an AutoBooting SCSI controller with an upgrade path to an '030 seemed like a reasonable design goal. This should all fit in a box no higher than the A500, and still allow you to pick up the Amiga without disconnecting the unit. The PCB would be a cheap double sided affair - no multi layer and no surface mount - to make assembly simple. To put all this on a small board would require extensive use of minimalism and GALs. What eventually came out of this was the SCRAM 500.

CASE

I think we must conclude that Commodore planned mischief when they designed the Amiga 500 Expansion Slot. This is apparent in the A590 and various other appendages which sprout from the left side of the machine. I don't think it is possible to make a box for the side of the A500 which really looks good. Who knows what they had in mind when they drew it up.

Hey Guys - Let's put the slot here and watch them get something onto that!

The SCRAM 500 lives in an unassuming, dignified rectangular extrusion - sturdy if nothing else - which pays no mind to bevelled snouts and mock plastic ribs and fins. True, it has been compared rather cruelly to a milk carton, but the extrusion provides good RFI shielding and simplifies assembly - the PCB with front and back just slides into the extrusion and is held by 2 screws.

The extrusion is wide enough to fit 2 cards. There is a companion card (which I am currently developing) with an '030 , 68882 and 8Megs of 32 bit wide RAM which can go in the case with the SCRAM board. I think this will make quite a desirable little expansion box for the Amiga 500.

AUTOCONFIG

The SCRAM 500 AutoConfigures as two boards - a 2/4/8 Meg RAM card and a SCSI interface card. These cards (notionally two - physically one)have their own product number and so on. The BERTIE Chip does most of the AutoConfig work by providing the sequences of config nibbles to the Amiga. The 74F521 (U19) initially decodes the AutoConfig space (E80000) via the R-NETS, and later decodes the IO base offset latched into U24. AutoConfig is inhibited by CFIN\ into U17A being high. At the end of the AutoConfig sequence the signal CFGO\ (Config Out for next board) is latched and also drives the "ACTIVE" LED on the front panel. This LED indicates that the Amiga has interrogated the SCRAM 500, found everything in order and included it in the IO board list. This is a very useful LED. It will immediately extinguish on reset, and only come on after a good config procedure.

Note that BERTIE looks at the RAM SIZE jumpers JP1, JP2 to tell the Amiga how much RAM is available (0/2/4/8 Meg). There is no latch for the RAM base offset, as it is assumed all the available 8Meg space is available. The Amiga always allocates RAM from 200000 onwards. If you want to put some other RAM in the space, partially populate the RAM array (2/4 Meg).

Even though the full 8 Meg space is decoded, the missing RAM chips can not drive the data bus, so something else can. You will get the odd wait state inserted by the refresh logic, but that is all.

PASS-THROUGH

The SCRAM 500 supports pass-through! So you don't have to forget CDTV and other devices. You may have the left side of your Amiga festooned with all manner of chunky boxes.

If you install the pass-thru option, the SCRAM 500 generates a ConfigOut signal for daisy-chaining to subsequent boards. Solder a jumper from pin 12 on the pass-thru connector to pin 11 on the SCRAM PCB. This will allow AutoConfigOut to daisy-chain on to the next device. The case has a hatch to allow access for pass-thru.

Be warned however, pass-thru can be a pain-in-the-pass-thru. You are connecting lots of things to the raw 68000 bus, so loading and noise can cause trouble. You are also relying on every device to follow the rules of AutoConfig. The SCRAM 500 will always configure RAM at 200000 and if you are using 4 Meg DRAMs then it will insert the odd wait state anywhere in the 8Meg RAM expansion space. These things are determined by the GALs, so if there is a conflict for some other expansion device it is possible to fix it.

I would recommend a HC68000 device in your Amiga if you are determined to pass-thru. This should give you cleaner signals with better noise immunity. Good luck.

RAM ARRAY

The SCRAM 500 uses ZIP DRAMs in the RAM section. As you can see from the PCB overlay, there are 16 ZIP locations in the top front corner of the board. You can use either

Continued on page 44

QUICK REFERENCE CARD FOR ED (the AmigaDOS text editor)

Keyboard Commands

Cursor Movement

Command	Action
RETURN	Split line at cursor and create a new line
TAB	Move cursor right to next tab position (does not insert a TAB character)
<up-arrow>	Move cursor up one line
<down-arrow>	Move cursor down one line
<left-arrow>	Move cursor left one character
<right-arrow>	Move cursor right one character
CTRL-E	Move cursor to top or bottom of screen
CTRL-I	Move cursor right to next tab position (TAB)
CTRL-M	RETURN
CTRL-R	Move cursor to end of previous word
CTRL-T	Move cursor to start of next word
CTRL-]	Move cursor to the end or start of line

Scrolling

Command	Action
CTRL-D	Scroll text down - half page
CTRL-U	Scroll text up - half page

Insert/Delete

Command	Action
BACKSPACE	Delete character to left of cursor
DEL	Delete character at cursor
CTRL-A	Insert line
CTRL-B	Delete line
CTRL-H	Delete character left of cursor (BACKSPACE)
CTRL-O	Delete word or spaces to right of cursor
CTRL-Y	Delete from cursor position to end of line

Miscellaneous

Command	Action
ESC	Enter extended command mode
CTRL-F	Flip case of character at cursor position
CTRL-G	Repeat last extended command line
CTRL-V	Verify screen (redraws the screen)
CTRL-[	Enter extended mode (ESC)

Extended Commands (press ESC first)

Cursor Movement

Command	Action
B	Move cursor to Bottom of file
CE	Move Cursor to End of line
CL	Move Cursor one position Left
CR	Move Cursor one position Right
CS	Move Cursor to Start of line
M number	Move to line number
N	Move to start of Next line
P	Move to start of Previous line
T	Move cursor to Top of file

Scrolling

Command	Action
B	Move cursor to Bottom of file
T	Move cursor to Top of file

Insert/Delete

Command	Action
A/line/	Insert line After current line
D	Delete the current line
DB	Delete the marked Block
DC	Delete Character at cursor
I/line/	Insert line before current line at cursor position
IF/filename/	Insert File filename at cursor position
J	Join current line with next (delete RETURN at end of current line)
S	Split a line at the cursor position

Find and Exchange (Search and Replace)

Command	Action
BF/string/	Backward Find search for string
E/str1/str2/	Exchange string1 with string2
EQ/str1/str2/	Exchange but Query first
F/string/	Find string - search forward
LC	Distinguish between Lower Case and Upper in searches
UC	Ignores Upper Case and lower in searches

Block Transfers

Command	Action
BE	Place a Block End marker at the cursor position
BS	Place a Block Start marker at the cursor position
DB	Delete the marked Block
IB	Insert copy of marked Block at cursor position
IF/filename/	Insert File filename at cursor position
SB	Show a marked Block on screen
WB/filename/	Write marked Block to filename

Save/Load/Exit

Command	Action
IF/filename/	Insert File filename at cursor position
Q	Quit Ed without saving file
SA	SAve text to current filename and continue editing
SA/filename/	SAve text to filename and continue editing
X	EXit Ed saving text to current filename

Tabs and Margins

Command	Action
EX	EXtend right margin
SL number	Set Left margin to column number
SR number	Set Right Margin to column number
ST number	Set number of columns between Tab stops

Miscellaneous

Command	Action
RP	RePeat a command until an error occurs
SH	SHow information on filename, tabstops, margins, block markers and buffer usage
U	Undo changes on current line
;	Separates commands on the same command line
( )	Group commands for purposes of repetition number Repeats following command number times

AMIGA

Public Domain Disks

Q-Fish Disks

1001 - 1030

*Please Note: The descriptions given below are from the **Fresh Fish CD-Rom** and are NOT the comments of the CCUGQ PD Librarians, the PD Compiler or the Cursor Editor.*

We were waiting for the new Fred Fish disks which were supposed to be produced in conjunction with one of the overseas magazines. These, however do not appear to be forthcoming, so an industrious club member, Neville Duguid has volunteered to produce some new "Fish" disks, sourced from the Public Domain material in Fred Fish's CD-ROM "Fresh Fish". We have called these disks "Q-FISH" and continued Fred's numbering system from 1001 onwards. Neville hopes to continue producing 30 or more new disks each month from the material in "Fresh Fish", and so keep Fred's tradition alive.

As I haven't got copies of these disks yet, I can't include pictures or screen-clips in this edition of Cursor, but I hope to do so with later disks.

Mike

Q-Fish 1031:

ButlerJames

This database program for address management is Hotkey activated. Very useful to avoid having to continuously enter an often used address into your favorite word processor, for example. Compatible with OS 1.2, 1.3 and 2.0. This is Version 3.2, an update of Version 2.4 on Fish Disk 856.

Kalender

A small but powerful calendar program. Version 2.2. OS 2.04+ required.

Xoper

Very comprehensive program to monitor and control system activity. Monitor cpu, memory usage, ports, interrupts and devices. Close windows, screens and show loaded fonts or last Guru code number. Clean up memory, flush unused libraries, devices, fonts. etc. and a whole bunch more! Spawns its own process. A very handy background task to have loaded. This is Version 2.4, an update of Version 2.2 on Fish Disk 318.

Q-Fish 1032:

AmigaGuide

Archive distribution of the AmigaGuide hypertext utility direct from Commodore. Contains developer examples and tools for Amiga Guide under V34/V37 and V39, plus a free print/sign/send-in distribution license for

AmigaGuide, amigaguide.library, WDisplay, and their icons. This is Version 34.6, an update of Version 34.3 on Fish Disk 920. Requires a decompression program, eg LhA on Q-Fish 1002.

CassLabel

CassLabel is a MUI based program for editing cassette and other types of covers. The GUI is fully font sensitive and resizesible, because of MUI, and supports user defined formats for output. Makes life easier (especially for the music freak). This is Version 2.1. Requires MUI.

Installer

Archive distribution of the Amiga Installer utility from Commodore. Contains V1.24 of the Installer, documentation and examples for developers to use when developing their software. The documentation has also been enhanced and brought up to date. This is Version 1.26, an update of Version 1.24 on Fish Disk 870. Requires a decompression program, eg LhA on Q-Fish 1002.

Q-Fish 1033:

chksum

Produces a checksum of a byte stream that should be the same as the standard SVR4 "sum" program. Note that the "sum" documentation is misleading as the checksum is NOT simply a 16-bit checksum of all the bytes. Version 1.0.

mg

A public domain EMACS style editor, that uses the GNU EMACS command set. Includes AREXX support. This is Version 3b, an update of Version Beta 2B+ on Fish Disk 352.

Zaphod

A binary file editor like FileZAP and NewZAP. Version 1.0. Requires OS 2.0+.

Q-Fish 1034:

bsplit

Split binary files into manageable pieces as specified by a byte count. Similar to the UNIX "split" program, but works with bytes, not lines. Version 1.0.

DeviceLock

A GUI interface for the CLI programm 'lock'. You can lock or allow writing to partitions via gadgets or hotkeys. You can configure it to your own needs and it should work with other (CLI-)lock programs, too. Version 1.2. Requires KS/WB2.0 (V37)+.

GraviSimu

Excellent Gravity Simulation Package. With this software you can simulate the behaviour of different bodies with different specs (masses etc.) like planets. Examples included. German version only. Translations welcome. Version 1.1.

PlaneView

Ray traced HAM ham animation of an airplane. Is 184 frames in 320x200x6.

Q-Fish 1035:

AmiCDROM

A CDROM disk filing system for the Amiga. It supports the IS O-9660 standard and the Rock Ridge Interchange Protocol. The CDROM drive is mounted as a DOS device (e.g. CD0:). This is Version 1.11, an update of Version 1.7 on Fish Disk 935.

MungWall

Munges memory and watches for illegal FreeMem's. Especially useful in combination with Enforcer. The output can go to either the serial or parallel port. Includes a new MungList program that examines used memory areas for MungWall tag info, and outputs a list of the owners of the various pieces of allocated memory, their sizes, etc. Can even identify the owner of the memory by task name. This is Version 37.64, an update of Version 37.58 on Fish Disk 800.

VirusZII

Release II of this popular virus detector that recognizes many boot and file viruses. The filechecker can also decrunch files for testing. The memory checker removes all known viruses from memory without 'Guru Meditation' and checks memory for viruses regularly. VirusZ has easy to use intuitionized menus including keycuts for both beginners and experienced users. This is Version 1.07, an update of Version 1.03 on Fish Disk 989. Requires OS2.O+.

Q-Fish 1036:

Genies_StarTeck

Seven 'must-have' genies for anyone with Professional Draw. These genies will increase your productivity, create new objects and facilitate special effects. The genies are PieCharts, Stars, Emboss/Engrave, ScalePercent, ZoomWindow, SelectByAttribute and GridSet. Well documented and includes installation script. Also includes a keyboard template, sample clip art and DTP tips and tricks. Version 1.0. Requires 'Professional Draw 3.0' or higher.

LHWarp

A program which will read tracks directly from your floppy disk, compress them using adaptive huffman encoding, and output them to a file. The resulting file can be used by lhwarp to reconstruct an image of the original disk. This is Version 1.40, an update of Version 1.21 on Fish Disk 318.

PowerSnap

A utility that allows you to use the mouse to mark characters anywhere on the screen and paste them somewhere else, like in the CLI or in a string gadget. PowerSnap recognizes all non proportional fonts of up to 24 pixels wide and of any height. This is Version 2.2a, an update of Version 2.2 on Fish Disk 968. Requires AmigaDOS 2.0+.

Q-Fish 1037:

Struct

Struct compiler, version 1.0. Struct is a programming language designed for speed. Register variables are an integral part of the language and are under the direct control of the programmer. Documentation and example programs included. Version 1.0.

SysInfo

Reports interesting information about the configuration of your Amiga, including some speed comparisons with other configurations, versions of the OS software, and much more. Binary only. This is Version 3.24, an update of Version 3.23 on Fish Disk 969.

Q-Fish 1038:

Flush

Flushes unused libraries, devices and fonts from RAM. Options include flush all, flush one type, report but don't flush, report which got flushed, and amount of memory regained. Runs from CLI. Version 1.2. Requires OS 2.04+.

PerfMeter

PerfMeter is the X's Performance Meter-style little meter, which shows CPU usage, CPU load, chip, fast and public memory, and uptime. The items can be selected from the Project menu. Version 2.2. Requires Kickstart 2.0+.

SuperDark

A screen blanker with some special features. It is similar to the AfterDark screen blanker in the PC and Mac worlds. Features include a lot of different screen effects via "modular" screen blankers, a screen locker, and more. This is Version 2.0a, an update of Version 1.5 on Fish Disk 858. Requires 2.04+.

Q-Fish 1039:

pdksh

A KSH-like shell that is compatible enough with the real UNIX ksh to be used with most scripts that the UNIX ksh can run. Version 4.5.

true

Trivially simple versions of "true" and "false" UNIX like shell commands, for cases where the shell in use has no equivalent builtin command. Version 1.0.

Q-Fish 1040:

AmigaFAQ

Lists some Frequently Asked Questions about the Amiga, and tries to give answers. Its intention is to help new users and to reduce the amount of news that most experienced users don't have time to read anymore. Sections on

Hardware, Software, Programming, Applications, Graphics and more. Formatted in plain ascii, AmigaGuide, DVI, and texinfo. Drawer also contains some useful text files on ftp sites, newsgroups, hardware tips and one on the history of the Amiga. Version 94.03.21.

Degrader
Degrades your machine to try to get badly written programs to work. Allows you to block memory, add non-autoconfig memory at reset, turn audio filter on or off, intercept privilege violation errors, switch off cache/burst modes and can slow down a fast machine. Also can swap the boot drive and force 50Hz or 60Hz. Will do things straight away, after one reset or after every reset. This is Version 1.30, an update of Version 1.00 on Fish Disk 562.

The S:Startup-sequence continued from page 9

Basic rules for changing these files are:
ALWAYS WORK WITH A COPY
Leave the SETPATCH as the FIRST or an early command.
MakeDIR before you ASSIGN logical devices to them.
Printout a copy of the files and work in pen first.
If in doubt leave the line in.
Actually IF IN DOUBT OUT, lets see what happens :-)

To get the printout.
place your printer "ONLINE"
with paper
open SHELL

```
cd s: <return>
type >prt: startup-sequence
<return>
(wait)
type >prt: startupII <return>
(wait some more)
```

This article will not go through the entire startup-sequence in detail. There is a program "DOSHelper" that lists all the commands in detail. This program is in the public domain and is useful as you go through your printout.

(NOTE: anything after a ; is a comment only. These are useful and provide a lot of information in the original files. If you are after more disk space they can be deleted).
Suggestions of lines that can go
`Echo` -these print to the screen,
-often useless

`FastMemFirst` -only used if you -have more than 1meg

`BindDrivers` -used if you have expansion devices the need for this cmd will be with the expansion hardware manuals, if you don't have the hardware then dump the command, if in doubt leave it in.

`SetClock` -no realtime clock? -then out it goes

`Resident` -some are REALY GOOD some a waste of space leave `resident CLI L:Shell-Seg SYSTEM pure add` leave `mount newcon` as these load the SHELL prog.

The other resident lines are mostly good for major CLI users, they take memory that might be better used someplace else.

`Mount -Speak/Aux/Pipe` if you don't use them.. out!

Aux might be good if you use the serial port BUT leave the Mount Newcon as mentioned above.

If you move what is left from StartupII to Startup-sequence, you may also delete the lines:

```
failat 11
run execute s:StartupII
wait >NIL: 5 mins
```

All this leaves you with something like:-

```
c:SetPatch >NIL: r
Addbuffers df0: 10
cd c:
FF >NIL: -0
resident CLI L:Shell-Seg SYSTEM
pure add
c:Execute pure
mount newcon:
makedir ram:t
assign T: ram:t
makedir ram:env
assign ENV: ram:env
makedir ram:clipboards
assign CLIPS: ram:clipboards
SYS:System/SetMap usa1
path ram: c: sys:utilities
sys:system
s: sys:prefs add
LoadWB delay
endcli >NIL:
```

To eXit ED and SAVE changes
press esc
then x
then <return>

I don't say this is the best startup-sequence, and for my system it would NOT be.

You have to decide what is best for your use and your system.

The worst that can happen is the system won't start .... just reboot from a copy of your original disks then try again.

You CANT BLOW THE HARDWARE - give it a go.

Enjoy the Machine!

CHRIS

Aliases I Have Used continued from page 13

habit of typing "emacs" when they should have typed "memacs". They would have put the alias in their Shell-Startups to make life easier for themselves.

Amiga owners who use MS-DOS machines at work can similarly improve the quality of their home life by using Shell commands like:

```
alias del delete
alias rd delete
```

Then, even if they are always typing 'del' when they mean 'rd' (to remove empty directories from their IBM's hard drive), it won't matter which they use on the Amiga.

Alias can also save you the need to copy rarely used commands from floppy or CDROM to your hard drive. You only need to remember their names, not which disk they are on, nor whether you copied them or not, if you use this incredibly simple technique:

```
alias <filename> <volume:path/
filename>
```

For example, you might include this alias in your Shell-Startup file:

```
alias StripANSI
Fish_and_MoreIII:FishDisks/776-
800/Disk_798/StripANSI/StripANSI
```

Then you could execute "StripANSI" at any time, by typing for example:

```
StripANSI Log.text par:
```

It does not waste time searching paths for something that may not be there, but remembers where to look immediately, and tells you which CDROM to insert if it is not already mounted in a drive. If you want to keep an active database of potentially useful PD software, what's wrong with your Shell-Startup? It is never more than a few mouse clicks away, and does not use any resources until you open a Shell.

The second example from your Shell-Startup goes a step further. It includes one of the Copy command's "keywords" as part of the alias.

```
alias xcopy copy clone
```

You can find any command's keywords by typing its name followed by a query "?" at the Shell prompt. For example, "`COPY ?`" will respond with:

```
FROM/M,TO/A,ALL/S,QUIET/S,
BUF=BUFFER/K/N,CLONE/S,DATES/S,
NOPRO/S,COM/S,NOREQ/S:
```

The important thing to note here is that "CLONE" is included in the list of keywords.

The "S" in "CLONE/S" defines "Clone" as a "switch". When included, it modifies the behaviour of Copy by forcing it to copy some of the information stored in the file's header along with the rest of the file. This includes the file's creation or last modification date. If you use plain "Copy" in lieu of "Copy Clone" (or "XCopy"), potentially valuable information about the original file is not passed on. The new file's date stamp would be nothing more than the date the copy was made. This could make a big difference when it comes to "disk sleuthing". I will have more to say about "XCopy" next month when I answer your Question 2. In the meantime, keep in mind that an alias definition may include a selection of keywords in addition to the command name.

I know of Amiga users who have spent much time (and sometimes money) chasing up extra commands like Flush, Touch, Dismount for their Amiga's C: or Tools directory. "Alias" combined with knowledge of the standard AmigaDOS commands, really pays off here.

```
alias Dismount Assign dismount
alias Touch SetDate
alias Flush Avail flush
          ; release unused memory
alias Say Echo to speak:
```

("Narrator.device", "speak-handler" and "translator.library" need to be in their appropriate drawers, with speak: mounted, before the last example will work).

The command arguments you supply when using (as opposed to defining) an alias, don't always finish up at the end of the final expanded command. They may be merged in at any point by including square brackets "[]" in the definition to specify an alternative insertion point. This allows:

(1) Access to ARexx functions:

```
alias hex Rx 'say d2x([])
alias dec Rx 'say x2d([])
```

(2) Filename manipulation:

```
alias man more
Work:Manuals/[].doc
alias guy Run MultiView
Work:HyperText/[].guide
```

(The last two examples allow you to keep all your software documentation together in a single directory. "`man lha`" will subsequently expand to "`more Work:Manuals/lha.doc`" when the alias is used).

(3) Who said you can't send ASCII control codes direct to your dot matrix printer from the Amiga Shell?

```
Alias CHR Eval to par: Lformat %c
alias PRINT Echo to par:
chr 14
print "This is enlarged."
```

(5) Or get a (reasonably) quick fix on a file that could be anywhere in a

hard drive partition...

```
Alias Whereis search Work: all file
Whereis VT ; for example
```

(5a) Or a list of eligible files and their locations...

```
Alias Whereis search all file
Work: pattern #?[]#?
Whereis .mod
```

(5b) Without superfluous ".info" files?

```
Alias Whereis search all file
DH1: pattern #?[]~(#?.info)
Whereis player
```

(6) You can even embed additional commands in the output of other commands by enclosing them in ‘backticks‘ (produced by using the key on the top left of your keyboard).

```
Alias bigprint echo "`eval 14
Lformat %c`[]" to par:
```

bigprint "This is enlarged or I'm not Epson-compatible."

We have finally reached the chaotic world of "Control codes" and "Escape sequences" which we need to confront before we can interpret the next three examples in your Shell-startup. This can appear like a real minefield to the unwary.

To see what I mean, try typing "alias" without any arguments now.

The listing it produces on your screen this time appears garbled because there is a command in the form of an ASCII Control Code embedded in the text. To the Amiga Shell, CHR 14 means "Switch to the ALTernative Character Set". Enter "Unalias bigprint" to get rid of the problem, then "Alias" again.

There is insufficient space to investigate that particular code right now. I have merely shown you the tip of a huge iceberg, so that you may be in a better position to appreciate what follows.

First of all, make sure you know what these next three example aliases in your Shell-Startup do. Try them out by entering them as commands at the Shell prompt.

```
clear   ; (clears the window)
reverse ; (changes the Shell to
"reverse" video)
normal  : (restores the window
to normal)
```

Maximise your Shell window to full-screen size, and issue the following commands:

```
type s:Shell-Startup
alias
```

The output of these two commands will allow you to contrast the alias command's arguments with the way they are interpreted and stored internally by AmigaDOS. Here are the two different versions of just one of the aliases you will see listed in your Shell window. Notice that the asterisks have disappeared from the alias command's own version, and a special graphics character appears in place of the 'E's in the Shell-Startup file.

```
alias clear "echo *"*E[0;0H*E[J*
                                " "
Clear              Echo "[[0;0H[[J"
```

(A more usual way to represent this in plain text would be "ESC [0;0H ESC [J", the blank spaces being introduced purely as an aid to readability).

I have actually cheated a bit here. It is not possible to reliably quote the graphics character that Alias uses to represent ESC, so I have substituted a reverse square bracket "[". This is the way it would appear on your input line if you entered it directly using the Amiga's Escape key. To confirm this, enter the following alternative alias:

```
alias
alias Clear echo [c
alias
;(note that the same graphic
character appears in this alias
as in all the others).
Clear
```

In the first of the two previous examples, there are two separate levels of Escape codes in use at the same time. What makes it even more difficult to explain is that many users already have incorrect assumptions about what "Escape" actually means in computer terminology. If this is your first look at AmigaDOS, Derek, you couldn't have picked a trickier place to start.

Most computer users believe that the Escape key is for "escaping" from programs, or backing out of inappropriate menu or function-key selections. This is believed by so many users, it has actually come true! In the MS-DOS world it is an almost universally adopted convention. (Not only does Microshaft program their computers - it also programs their minds!) Many Amiga programs, "MuchMore" for example, also allow you to quit by pressing the Escape key. Who was it who said that a lie, repeated often enough, becomes the truth?

There is no harm in this alternative use of "Escape" other than making the original (and still very important) meaning less clear.
The original purpose of the 'Esc' key was to generate Code 27 (ASCII 'ESC'), in the same way that the 'A' key generates Code 97 (ASCII 'a'). The difficulty is in representing ESC (and other control codes) on the printed page or screen. Control Codes are really embedded commands which, depending on the device receiving them, are either obeyed instantly, or

completely ignored. Unlike the AmigaDOS command embedded in backticks above, ASCII Control Codes are almost universally recognized and are acted on regardless of the textual context in which they occur. Trying to quote Ctrl-L directly in a text document, for example, will cause the printer to eject the page at that point, or skip to the next page. There is no way to stop it doing that other than using different characters to represent the code when you refer to it in print. (Maybe that is why no one tackled your question last issue, Derek. It is a difficult topic to discuss coherently in print).

The designers of ASCII knew that the 32 available control codes (0-31) were not enough to cover all printer and miscellaneous equipment functions, so they deliberately left a "hole", Code 27, through which their standard definitions could be "escaped". This code, labelled 'ESC' should be interpreted as "Don't print the next character, but treat it as an extra control code." This extra command could be defined by an equipment manufacturer to perform functions possibly unique to his own particular hardware. Among other things it could imply a fixed length other than one character, i.e., how many following characters are not to be printed as text, but treated as part of the "Escape sequence". (Some printers receive entire graphic screen dumps embedded in escape sequences).

Thus you could conceivably have a missile launcher command "ESC PUKE", which might mean "Point Up at Klingon Empire", or anything else you choose. The only essential requirement is that both the sending and receiving equipment agree on the same definition of what it means. The missile launcher might be a totally unique piece of equipment, but by utilizing ESC rather than re-defining other codes, other standard pieces of equipment on the site (eg digital readouts) could also be controlled via the same channels. The problem for us is that we can't tell by looking at it where the escape sequence ends, or whether the space following "ESC" is there for readability - or a literal part of the command. Or perhaps it could all be just printable text in a document telling the operator how to use Escape codes. Remember, the real live commands can be embedded in text. At some of its many levels, the Amiga also works this way.

ANSI is a code which defines, among other things, standardised Escape sequences for the most commonly used terminal and printer functions. This has been incorporated into the day-to-day language of the Amiga Shell. Type 'Alias' on its own, to see some examples listed in your Shell window.

Remembering that I have substituted reverse "["s for the graphic characters representing ESC in the console window, here is one of your other examples in that format:

```
alias reverse echo
"[[>1m[[30;41m[[0;0H[[J"
```

You can enter it directly as shown, but if you want to incorporate it in a script, it is highly recommended that you follow the same form as used in your Shell-Startup file. This will avoid the sort of complications that embedded control codes can cause. Commands like Echo, which may have literal strings among their arguments, have problems of their own when certain characters are embedded in text. How would you include a Newline character in a command that is terminated by the end of line for example; or a quote in a string that is delimited by quotes?

For this purpose, AmigaDOS implements the asterisk "*" as a much more localised "escape" code of its own. Provided the string argument is in quotes, *N represents Newline, *" allows you to include an internal quote, and ** inserts a literal asterisk. You may be familiar with the backslash "\" used in a similar context elsewhere. Note that neither of these characters necessarily has anything to do with ASCII ESC. However, since ESC is the normal way of introducing the ANSI Escape Sequences supported by the Amiga Console, "*E" has been pressed into service as a way of representing ESC in AmigaDOS commands. In most of these (also known as Amiga "Console Codes") the combination ESC[(and thus "`[[`" or "`*E[`") is treated as a single logical unit called the CSI or "Control Sequence Introducer".

With this in mind, we are finally ready to have a go at interpreting your "Reverse" alias. Using the four CSI's as starting points, we can now split the control string into its four logically separate commands. It then becomes simply a matter of looking them up to see what they mean.

Did I say "simply"? the codes are defined in the standard Amiga header files Include:devices/console.h and include:devices/printer.h, and explained on pages 67-73 of the Addison-Wesley Amiga ROM Kernel Reference Manual: Devices (Third Edition).

In the early days of the Amiga,

books about programming the Amiga often re-printed a list of "printer codes" in an appendix without making any attempt to clarify them or explain what they meant. These books are becoming out of date and hard to find - although some are probably still in the CCUGQ Library. From memory, one of the best is "AmigaDOS Inside and Out", Abacus (1990 Revised Edition). Not all the printer codes are relevant to the console, but you can usually work out which ones apply by studying their definitions and/or trying them out.

The Console Codes of most concern to us here are:

ESC[x;yH	Set Cursor Position
ESC[J	Erase to End of Display
ESC[...m	Select Graphic Rendition

The x and y represent decimal numbers (ASCII digit strings) and the three dots represent any number of position-independent options separated by semi-colons.

Armed with this information, it is now easy to see that Echo "[[0;0H" means Set Cursor Position to location 0,0 (Home the Cursor), and "[[J" (Erase to End of Display) from that location clears the whole window. When you think about it, "Alias" is a natural for this sort of thing - you wouldn't want to type it from scratch every time!

Your next example:
Echo "[[>1m[[30;41m[[0;0H[[J" ("Reverse"), also terminates with the same two commands that clear the window. The two lower-case 'm's tentatively identify the other two commands in the string as instances of "Select Graphic Rendition".

Here is a brief summary of the other codes that may be included in "Select Graphic Rendition":

0-8	Attribute ON
22-28	Attribute OFF
30-37	Set Character Foreground Colour
39	Reset Char-Cell (f/g & b/g) Colours
40-47	Set Char-Cell Background Colour
>0-7	Set Window (global background) Colour

The low digit of each colour parameter represents one of the first eight Amiga Colour Registers as determined by Preferences. So Echo "ESC[32;47mX" would set text to the same colour as Register 2, and its local background to the same as Register 7. It would also print the following literal "X" (not part of any escape sequence) in those colours. The Shell will continue using that colour scheme until you change it. If your Workbench Screen is set for only four colours, Registers 4-7 will be the same as Registers 0-3. Don't forget to use the Escape-key rather than typing 'ESC' literally when you type in this example.

The "attributes" set by digits 0-8 (and cleared by prefixing a "2") are defined for Kickstart 2.0 as follows:

0	Plain text	
1	Boldface	
2	Faint (halfbrite)	22Normal (not bold)
3	Italic	23Italic OFF
4	Underscore	24Underscore OFF
7	Reverse	27Reverse OFF
8	Concealed mode	28Concealed OFF

Echo "amESC[7miESC[27mga" prints a text string containing two embedded commands. It should print "amiga" with the "i" in reverse video. Again, don't forget to use the Escape key. (You can probably see how adding a space after 'ESC' in this example would reduce rather than improve readability - quite apart from causing doubt as to the string's exact composition. You really do need to know some of the theory behind it when typing in an example expressed in that format).

'Set Graphic Rendition's optional parameters may occur in any order except for the >0-7 parameter which must be last (directly preceding the terminating "m"), if it is used.

Looking at 'reverse's definition once more, it is obvious that whoever wrote it wanted the background colour set prior to making the other changes.

```
echo "/[>1m/[30;41m/[0;0H/[J"
```

To do that, he had to split it into two commands, otherwise the ">1" would have had to go last. I am sure you will be experienced enough by now to recognize that the "/" characters in the example are included here to represent the Escape code. (Reverse-[normally behaves too unpredictably to risk including it in something you send to the printers - unless you have made some prior special arrangements).

Note that quoting the whole Echo (or other) command in an alias necessitates prefixing asterisks to any quotes already surrounding the quoted command's arguments. Such quotes are necessary if the argument contains spaces, semi-colons or codes requiring asterisk-prefixed characters to represent them. For clarification, compare the on-screen output of the "alias" and "type s:Shell-startup" commands again.

This brings us to the final command

in your Shell-startup:

```
Prompt "%N.%S> "
```

It contains three literal characters ".", ">" and " ", and two format specifiers similar to the %C used in the "eval" command example above. "%N" represents the Shell's process number. If you are running more than one shell, this will produce different results for each Shell. The "%S" is the AmigaDOS path to the Shell's current directory. You could experiment by selectively excluding these items from your own prompt.

```
Prompt "%N.> "
```

You may also personalize the prompt by including things like your name, or add ANSI Escape sequences (Console Codes) using Escape-[(eg "*E[") as explained above.

Here is one for you to try. Note that blanks are always interpreted literally in strings that employ "*E", this being one of the advantages of using that format.

```
prompt "*E[32m%N.‘version‘:> *E[31m"
;(Note: the single quotes are backticks, not apostrophes; also it does appear not to work entirely as expected under WB 2.0)
```

I have heard people express dissatisfaction with AmigaDOS, claiming it is not as versatile as certain other DOS command languages. I can only conclude they have never investigated it fully. Come to think of it, I don't know many people who use it a great deal. I suspect that if Microshaft were to bring out a version of AmigaDOS for Windoze, we would never stop hearing about how versatile and powerful it truly is.

Amigas History continued from page 19

with management over the design choices they forced on us. The whole story is just too sad to tell and too long. But, in many of our minds, the A3000 (and A3000+) were the last real "desktop" Amigas that were engineered. (The A4000T is not bad motherboard but it needs a better CPU interface or at least a better CPU card with local RAM and it really should have had a DSP...).

Oh, just as a side note: The A3000+ design was such that it would have cost almost the exact same to build as the A3000 even with AGA and the DSP! But, that does not matter since the A3000+ was never finished or shipped. (there are a few prototypes that were made, and two of them even have the DSP working...)

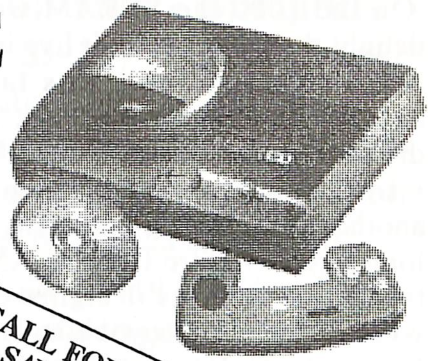
-- **Mike Sinz**

PARCOM PTY LTD

FOR ALL COMMODORE AND AMIGA WARRANTY AND NON-WARRANTY REPAIRS

AMIGA CD 32 CONSOLE

- * 32 BIT POWER
- * 16.8 MILLION COLOUR PALETTE
- * PLAYS MUSIC CD'S
- * 256K MAX COLOURS ON SCREEN
- * 2MB RAM
- * MANY TITLES
- *LATEST RELEASES



CALL FOR $$ SAVINGS

SPECIAL CONCESSIONS TO CCUGQ MEMBERS ON PRODUCTION MEMBERSHIP CARD

SPECIALISING IN AMIGA MODIFICATIONS & REPAIRS

Computers - Disk Drives - Monitors

*** NOW AT THE ONE ADDRESS ***

TECHNICIANS
Anthony Bamberry & Bradley Jesse

WHITESHILL
SHOPPING VILLAGE
SAMUEL STREET
CAMPHILL

PHONE: 843 2120

FAX: 398 4832

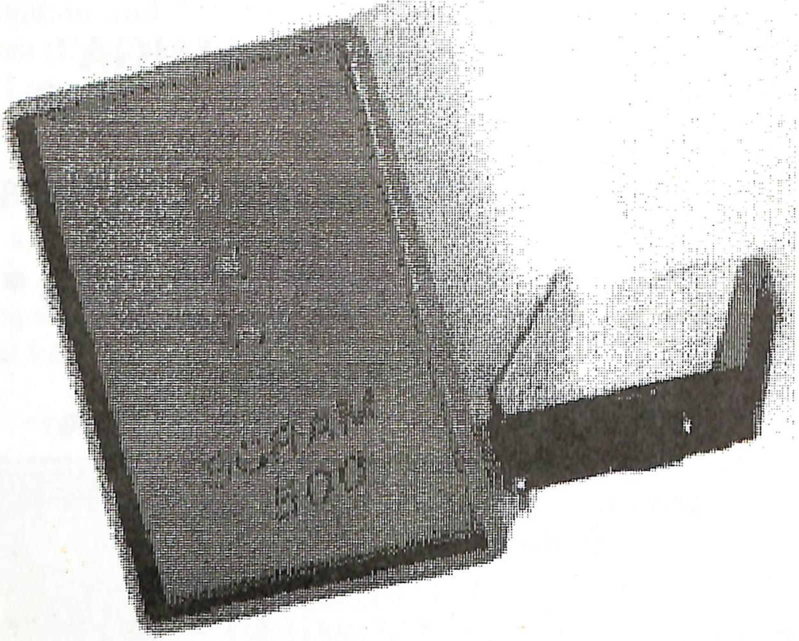
SCRAM500 continued from page 21

dimensions. I switched the Amiga off, walked to another part of the house. When I returned and restarted Ami she STILL had 2.5megs. It was many months before I got the harddrive and that again is another story for now I had achieved the impossible.

On the 1000 the SCRAM was a delight though it couldn't live in its A500 case and the lights faced backwards, I began to dream. The dreaded computer affliction had visited, I could upgrade. Herein lies another story but suffice to say not long passed before I had an A500, the SCRAM in its designer case with the little lights winking at me from in front. It had started up first time on this machine though I have now moved over the newer processor. The whole setup has proved to be robust and reliable, Fast ram IS faster and three megs is more memory than I have yet to use. The SCRAM500 looks good and goes better and was really not hard to build, though the first few smoke tests are a worry. I have to thank Mark Ireland at MegaMicro (02 980 6986) for a terrific product and great user support. I will let you know the rest of the A500 story another time.

Enjoy the Machine.

.... **CHRIS**



FOR SALE

FINAL COPY II **$90**
HEIMDALL **$20**
ZOOL AGA **$20**
DRAGONS LAIR II
("TimeWarp") **$20**

All are in good condition with original boxes, disks and manuals.

Damien Crabtree
(07) 805-1419 (A/H)

FOR SALE

COMMODORE 1940
MULTI-SYNC MONITOR
Ideal for Amiga 1200 or Amiga 4000
Suits all AGA Modes
Perfect Condition
9 months old
With Box and Manual
$450

Brian Hay
(07) 378-5595

This space is available for Club Members private advertisements of computer hardware or original software. Advertising here is FREE.

For Sale - Bargains

Star SJ-144 Thermal Colour Printer. Perfect condition.
REDUCED TO **$695**
2-meg ChipRam Board incl. Ram & Agnes chip. Suits Amiga 500 or 2000 ....**$195**

All are in good condition with manuals.

Keith Antoine
(07) 300-2161

FOR SALE

AT Bridgeboard (A2286) with a 5.25" F.D.D....**$150**
80287 Maths co-processor............**$50**
Paradise VGA Plus 16 VGA display card **$50**

Contact Jack Oudyn
(07) 821-3206 (Home)
(07) 390-2455 (Work)

SCRAM500 Design continued from page 23

1Meg (256Kx4) chips or 4Meg (1Mx4 chips) as they are pin compatible. The standard BERTIE chip only allows for 2/4/8 Meg, so partial population with the older 256Kx4 RAMs is not possible. These days 2 Meg is pretty cheap, so I can't see that much call for less than 2 Megs. Using 4 Meg chips the minimum configuration is just 4 ZIPs for 2 Megs!

You will notice that the RAM circuit does not use buffers. The design goal for the SCRAM 500 was small and simple - which didn't really allow room for buffers. A few years ago I designed a 2Meg RAM board for the Amiga using a similar circuit. Nearly 10,000 of these boards were made and functioned reliably. What I did discover was that some (most?) Amiga 500s don't have pullups on their data bus.

There are holes there for them, but they aren't populated. When the SCRAM 500 was being debugged it was observed that with NO RAM installed, the SCSI interface was prone to errors during long transfers. Plugging RAM in fixed the problem - which seemed to indicate that the Amiga 68000 was prone to noise on the data bus. 10K pullups installed on the 68K data bus fixes the noise problem - I think they should be on the Amiga 500 main board as standard.

RAM CONTROLLER

The RAM controller is probably the most complex section of the SCRAM 500. U20 - U22 (LS257) are address multiplexors which switch the bus address to RAS and CAS addresses. GRISWOLD is the main timing generator which converts a 68000 cycle into RAS and CAS cycles, and inserts a refresh cycle every 16uS or so. HUMPHREY converts memory addresses into Bank RAS pulses, accomodates 1M/4M chips and handles CPU XRDY Timing. The HCT393 counter (U26A) divides the CPU E clock by 12 to give the refresh rate.

The SCRAM design uses CAS before RAS refresh for the DRAMs. This is an economical design because you don't need a refresh address counter - the DRAMS have this built in. GRISWOLD generates a RDY signal to hold off any 68000 cycle which starts whilst a refresh cycle is in progress.

Pushing most of the DRAM control functions into GRISWOLD and HUMPHREY results in a simple, compact design. This modular design is also easy to fix. The LS257 chips are pretty tough - I don't think anyone could blow them up - which leaves GRISWOLD, HUMPHREY, and the RAM chips (all socketed). If you poke around this circuit with a scope, you will soon see what is going on.

SCSI INTERFACE

If you look at the SCSI controller schematic (sheet 2 of 3), you will see it is also very simple! CYRIL 8, DP8490, Terminators and SCSI connector, yet this interface returns DiskSpeed results of over 500KB/sec with a standard Amiga and well over 1Mb/sec in an 030 machine. The key to this compact design is using a single PLD (CYRIL 8) to handle all the glue functions for the SCSI chip. This includes address decode, LED controller, BOOT ROM mapping, interrupt and DMA synchronization.

The data transfer method used in the SCRAM is known as DTACK SYNCHRONIZATION. A lot of designers use it - I first saw it in a Motorola APP NOTE. All it does is delay 68000 bus cycles until the SCSI chip has the next byte to send. In general, this is not a good idea for any DMA device, but for high speed block orientated things like SCSI, it is fine. The only thing about this type of interface is you need tight, optimized code to make it perform.

The alternative to DTACK SYNCHRONIZATION is a DMA CONTROLLER like Commodore uses. DMA is far more complex to design, and uses up a lot more board space - unless you have access to Gate Array technology. The advantages of DMA are not that apparent to a user, and for an Amiga 500 system hardly justify the additional cost and complexity. Anyway, I like minimal designs.

PROGRAMMING MODEL for the SCSI CONTROLLER

The SCSI part of the SCRAM 500 AutoConfigs as a separate PIC from the RAM controller. This section comes up as a 64K IO device with a switchable vector to ROM boot code. The SCRAM will usually be the only AutoConfig device attached, and so the IO block will be allocated offset E90000, directly after the AutoConfig space of E80000.

```
Memory Map
                ----------------  |
|      |   Boot ROM    |          |
|    ----------------  C000h      |
|       |  Pseudo DMA   |         |
Data Transfer |   ----------------
8000h      |  SCSI Registers |
----------------  6000h        | Set
LED ON     |       ----------------
4000h    |            |     |  Boot
ROM    |    |         |        --
--------------  0000h <-- Offset int0
64K IO space (E90000)
```

0000 - 3FFF BOOT ROM available in even bytes. 4000 - 5FFF Write in this space sets LED on. 6000 - 7FFF SCSI chip in even addresses. Clears the LED 8000 - 9FFF Direct data access (even bytes) C000 - FFFF BOOT ROM available in even bytes.

The BOOT ROM is available at both 0000h and C000h in the IO space at even addresses. This is the driver code which is read in by KickStart at Boot time.

The SCSI chip is located at offset 6000h in the IO space. Only even locations, so valid addresses are 6000h, 6002h, 6004h, 6006h, 6008h, 600Ah, 600Ch and 600Eh.

To set the LED on, perform a write to location 4000h in the IO space. The LED is cleared by any access to the SCSI chip (6000h).

The SCRAM 500 supports a Direct Data mode of transfer to improve disk throughput. The direct data access mode is a form of psuedo DMA. Instead of hardware doing the data transfer, Data is read or written by accessing location 8000h which - via CYRIL 8 - directly accesses the DMA port on the SCSI chip.

CYRIL 8 automatically synchronizes the 68000 to the SCSI controller DMA Channel via DRQ and XRDY so you don't have to do any status checking. Just read or write. The routine which does this transfer should really be organized as a block sized subroutine (512 bytes). Between blocks you should check status to ensure there is a valid REQ\ from the device. This is because if you initiate the block transfer (access 8000h) with no DRQ pending, the 68K will be held off indefinitely by XRDY. If there is no data pending from the SCSI controller, or the SCSI terminates unexpectedly, then the system could hang. Make sure that if you ever access 8000h (Direct Data access) you know FOR SURE you have a data block waiting for you with DRQ active.

The unpleasant situation mentioned above is affectionately known as a "DTACK hang". CYRIL 8 has a failsafe mechanism buit in to protect against non-self-inflicted DTACK hangs caused by unforseen terminations in the SCSI chip. If the SCSI Interrupt is set, this signal disables DTACK synchronization - releasing a DTACK hang. Otherwise, the interrupt is passed straight to the Amiga from the SCSI chip as INT2\.

You can use any programming technique to minimise the overheads of hauling in the data from 8000h. The best method is probably to use MOVEP IO accesses and long word transfers. Do a cycle count and you will find it is possible to get a 33% increase in throughput. Also note that low address lines are not used in decoding the 8000h space, so 8000, 8002, 8004 etc. will all work so you could treat the Direct Data space as a block of 512 WORDS, where the even BYTES are valid data.

Norman Jackson

MEGADISC
Subscriptions Currently available through CCUGQ
This is the ORIGINAL Australian Disk Magazine for the AMIGA
Subscribe Now!!

Please submit articles on disk in ASCII format (if possible) and post them to :

Mike Williams
32 Bulwarna St
Shailer Park Qld 4128
or upload to:
The Cursor Editor
CCUGQ BBS
(07) 300-6357 or (07) 805-5928

AMIGA PUBLIC DOMAIN DISKS

Our extensive range of over 2000 Amiga P.D. disks is available at:

1. Our Workshop Meeting
2. Our Main Meeting
3. By Mail
4. On our BBS (a selection).

Copying of P.D. disks by Club Members at our Workshops is free.
Disks can be purchased at our Main Meetings for $2 each.
Requests for disks by Mail should be directed to The Secretary, enclosing a cheque/money order payable to C.C.U.G.Q.
Cost is $2 per disk plus $3 postage for each 6 disks or part thereof.
P.D. Librarians are Ron and Warren Blake Phone (07) 287-1790.

MEMBERSHIP SCHEDULE

ANNUAL FEES

Ordinary Membership	**$40**
Pensioner Membership	**$30**
Country Membership	**$30**
Library Administration Fee	**$10**

(Required for access to Commercial, Public Domain and Magazine Libraries)

COMMODORE COMPUTER USERS GROUP Qld.

P.O. BOX 274 SPRINGWOOD QLD. 4127.

MEMBERSHIP APPLICATION FORM

Date of application : / /

Name :

Address :

Postcode :

Phone (Home) : **(Work) :**

Business Name (if applicable) :

Type of Computer(s) :

Printer Make/Model :

Modem Make/Model :

Other Equipment :

Computer Knowledge (Languages/Technical etc) :

Area of Computer Interest :

Please find enclosed Cheque / Money Order for $

Payable to C.C.U.G.Q. Inc.

Type of Member (see over) :

Signed :